Claims

1. A method of reducing the amount of computer memory utilised in calculating a formula on a collection of series of data values, the method comprising the steps of:

(a) for each data value member of a first one of said collection, determining a window around a current data value member of data values required to calculate said formula;

(b) utilising said window to determine data values to be stored in computer memory when calculating the formula when applied to other series of data values in said collection.

2. A method as claimed in claim 1 wherein said step (b) comprises, for members within a series of data values,

(a) for a first one of said members, utilising said window to determine a first initial set of data values in the series to be stored in a portion of computer memory; and

(b) for subsequent current members, loading the data values into the same portion ofcomputer memory over locations over previously loaded data values.

3. In a computer system with a primary memory store, a method of carrying out a formula calculation on a series of data values, the calculation being carried out using members of the series with the calculation for a current member of the series being dependant on other data values located relative to the current member of the series, the method comprising the steps of:

(a) for a given current member of said series, determining from the formula a relative series of consecutive data values required for determining said formula for said current member;

(b) for each current member of said series of consecutive data values:

(i) ensuring a corresponding relative series of consecutive data values to said current member are currently loaded into said first primary memory store;

(ii) performing said formula calculation to determine a current output value.

4. A method as claimed in claim 3 wherein said step (b) comprises:

for a first one of said current members, loading into said first primary memory store and initial corresponding relative series of consecutive data values; and

for subsequent current members, loading only new members of said corresponding relative series into said primary memory store.

5. A method as claimed in claim 4 wherein said series of consecutive data values are arranged in a column.

6. A method as claimed in claim 3 wherein said method is applied to multiple groups of consecutive data values using a different formula for each group.

7. A method as claimed in claim 6 wherein new values are loaded into said primary store at an address determined by a modified modulo arithmetic operator which produces positive address values only.

8. A method of reducing the amount of computer memory required to be utilised in the calculation of a formula applied to a collection of series of data values, the method comprising the steps of:

(a) for a first member of said collection:

determining a minimum window of data values required for storage in computer memory for the calculation of said formula when applied to said first series;

(b) for subsequent members of said collection:

utilising the minimum window to determine data values of a series that need to be loaded into said computer memory for calculation of said forumla.

9. A method as claimed in claim 8 wherein said step (b) further comprises, moving the series of data values through a fixed portion of computer memory, and utilising the data values whilst in computer memory to calculate said formula.

10. A method as claimed in claim 9 wherein said step (a) further comprises:

calculating said formula on an initial data set to determine a minimum window size, the determination being by the steps of providing an initial current window around a current data value to be calculated; determining in accordance with the formula, other data values required,

and if the other data values are below the current window then: sliding the window down or making the sliding window bigger to encompass the other data values, including if the present sliding direction is to slide up, then making the window bigger to encompass the data values, and if the window becomes large enough to encompass the data values, setting the sliding direction to be undefined; if the present direction is sliding up then sliding down by dropping previously calculated values from the top of the current window, provided no value that is busy being calculated may be dropped off; upon completion of the calculation of the formula, determining a final required window size; subsequently, for each of the other data sets in the series, utilising the final window size to calculate the formula for each data set, moving through the data set element by element storing values in the current window within primary memory.